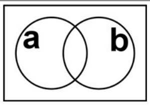
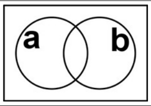
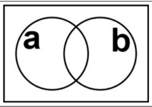
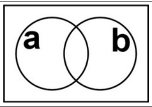
Nand, Nor & De Morgan

A ⟹ B

1. ***Praktische Negationen***

NAND- und NOR-Operator (Junktor) kennst du bereits aus dem Informatikunterricht. Ins Deutsche könnte man sie zum Spaß mit NUND und NODER übersetzen. Für beide Verknüpfungen gibt es aber trotz ihrer Bedeutung keine deutschen Wörter, wir greifen daher auf ihre Definition als Negation der Konjunktion bzw. Disjunktion zurück. Fülle die Wahrheitstafel aus und färbe die zugehörigen Venn-Diagramme passend.

a	b	a ˄ b	¬ (a ˄ b)	a ˅ b	¬ (a ˅ b)
Venn-Diagramme					
Logikgatter		*a AND b*			

<u>*Schreibweisen*</u>

*"**a NAND b"** wird nach Henry Maurice Sheffer (1882-1964) auch als "Sheffer-Strich" (**a|b**), "Sheffer-Pfeil" (**a🔧b**), als* ¬ **(a˄b)** *oder* $\overline{\boldsymbol{a\wedge b}}$ *geschrieben. Die NOR-Verknüpfung wird nach Charles Sanders Peirce (1838-1914) auch als Peirce-Funktion bezeichnet. Als Schreibweisen sind neben **a NOR b** unter anderem auch* **a b** *,* ¬ **(a˅ b)** *oder* $\overline{\boldsymbol{a\vee b}}$ *gebräuchlich.*

2. **NAND & NOR** (Partnerarbeit)

Peirce wies nach, dass man mit dem NAND (oder NOR)-Junktor alle logischen Verknüpfungen darstellen kann. Das hat große Bedeutung für die Praxis, da man so alle (!) Logikschaltungen jeweils mit nur einem Gattertyp herstellen kann. Aus Klasse 9 wissen wir bereits, dass die drei Basisjunktoren ¬, ∧, ∨ ausreichen, um alle logischen Verknüpfungen darzustellen. Hier sieht man nun, dass dazu sogar der NAND-Junktor ausreicht[1]:

NAND			NOR	
Umformungsschritt(e)	*Regel*		*Umformungsschritt(e)*	*Regel*
$\bar{a} \Leftrightarrow \overline{a\wedge a}$		¬ *(NOT)*		
1. $a\wedge b \Leftrightarrow (a\wedge b)\vee(a\wedge b)$ 2. $\Leftrightarrow \overline{\overline{(a\wedge b)\vee(a\wedge b)}}$ 3. $\Leftrightarrow \overline{\overline{(a\wedge b)}\wedge\overline{(a\wedge b)}}$ [2]		∧ *(AND)*		
1. $a\vee b \Leftrightarrow (a\wedge a)\vee(b\wedge b)$ 2. $\Leftrightarrow \overline{\overline{(a\wedge a)\vee(b\wedge b)}}$ 3. $\Leftrightarrow \overline{\overline{(a\wedge a)}\wedge\overline{(b\wedge b)}}$ [1]		∨ *(OR)*		

a) Analysiert die Umformungen der linken Seite und begründet jeden Schritt mit einer Regel.

b) Führt rechts analog die Basis-Verknüpfungen ¬, ∧, ∨ auf den NOR-Junktor zurück.

1 Statt ¬(a˄b) wird hier die Schreibweise $\overline{a\wedge b}$ verwendet, um die Negationen besser zu visualisieren.
2 Mit dem "Sheffer-Pfeil" geschrieben: Konjuktion a˄b=(a↑b)↑(a↑b) ; Disjunktion a˅b=(a↑a)↑(b↑b)

3. ***In gewohnter Weise***

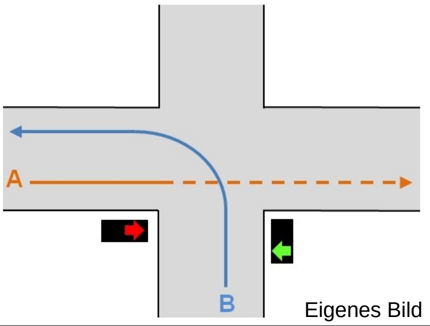
Notiere die Umformungen aus Aufgabe 2 *mit dem gewohnten Zeichen "▲" für die Negation, um die drei Basis-Verknüpfungen ¬, ∧, ∨ nur mit dem NAND- bzw. NOR-Junktor darzustellen. Bei Bedarf kannst du bei den entsprechenden Rechengesetzen nachschlagen.*

NAND	Nr.		NOR	Nr.
$\neg a \Leftrightarrow \neg(a \wedge a)$	(9')	¬ *NOT*	$\neg a \Leftrightarrow$	(9)
1. $a \wedge b \Leftrightarrow (a \wedge b) \vee (a \wedge b)$	(9')	∧ *AND*	1.	(9)
2. $\Leftrightarrow$	(10)		2.	(10)
3.	(11)		3.	(11)
1. $a \vee b \Leftrightarrow$	(9')	∨ *OR*	1.	(9)
2.	(10)		2.	(10)
3.	(11)		3.	(11)

4. ***Ampelschaltung***

Diese Aufgabe aus Klasse 9 liefert nun weitere Anknüpfungspunkte.

a: Ampel für Fahrspur A zeigt grün.
b: Ampel für Fahrspur B zeigt grün.



Eigenes Bild

a	b	*sicher?*
0	0	
0	1	
1	0	
1	1	

Sichere Terme:

a) Die Szene passt zu Zeile 2. Trage in der letzten Spalte ein, ob die Schaltung sicher (=1) bzw. unsicher (=0) ist. Welche Verknüpfung wird durch die Wahrheitstafel dargestellt?

b) Eine sichere Ampelschaltung könnte hier z.B. mit der Schaltungsgleichung $s \Leftrightarrow \neg(a \wedge b)$ *(bzw.* $s = \neg(a \wedge b)$ *) beschrieben werden. Interpretiere die Gleichung.*

c) Stelle weitere Aussagenterme auf, die für diese Kreuzung eine sichere Schaltung garantieren. Nutze dabei auch Äquivalenzumformungen wie die De Morganschen Regeln. Wie viele verschiedene Schaltungsgleichungen gibt es hier?

5. ***Kreuzung gesucht!***

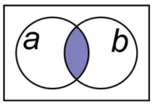
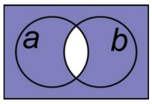
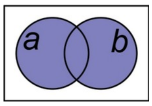
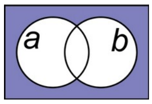
Für eine andere Kreuzung liefert die Aussage $s \Leftrightarrow \neg((a \vee b) \wedge c)$ *eine sichere Schaltungsgleichung, die ebenfalls mithilfe der NAND-Verknüpfung formuliert wurde.*

a) Analysiere, welcher Fall durch die Gleichung ausgeschlossen werden soll und skizziere eine passende Kreuzung. Stelle auch die zugehörige Wahrheitstafel auf.

b) Gib möglichst viele weitere sichere Schaltungsgleichungen für diese Kreuzung an.

Lösungen

1. **Praktische Negationen**
   Die Konjunktion a˄b und die Disjunktion a˅b werden gemäß der Tabelle negiert. In den Venn-Diagrammen sind die Zusammenhänge zusätzlich visualisiert. Dabei entspricht jede Teilfläche einer Belegung der Aussage-variablen in den vier Zeilen.

a	b	a ˄ b	¬ (a ˄ b)	a ˅ b	¬ (a ˅ b)
0	0	0	1	0	1
0	1	0	1	1	0
1	0	0	1	1	0
1	1	1	0	1	0
Venn-Diagramme					
Logikgatter		*a AND b*	*a NAND b*	*a OR b*	*a NOR b*

2. **NAND & NOR**
   Die drei Basis-Verknüpfungen ¬, ˄ , ˅ (und damit alle logischen Verknüpfungen) können mit nur einem Junktor dargestellt werden, wie in der folgenden Tabelle für den NAND- bzw. NOR-Junktor nachgewiesen wird:

NAND			NOR	
Umformungsschritt(e)	*Regel*		*Umformungsschritt(e)*	*Regel*
$a \Leftrightarrow a \wedge a$, daher $\bar{a} \Leftrightarrow \overline{a \wedge a}$	(9‘)	¬ NOT	$a \Leftrightarrow a \vee a$, daher $\bar{a} \Leftrightarrow \overline{a \vee a}$	(9)
1. $a \wedge b \Leftrightarrow (a \wedge b) \vee (a \wedge b)$	(9‘)	˄ AND	1. $a \wedge b \Leftrightarrow (a \vee a) \wedge (b \vee b)$	(9)
2. $\Leftrightarrow \overline{\overline{(a \wedge b) \vee (a \wedge b)}}$	(10)		2. $\Leftrightarrow \overline{\overline{(a \vee a) \wedge (b \vee b)}}$	(10)
3. $\Leftrightarrow \overline{\overline{(a \wedge b)} \wedge \overline{(a \wedge b)}}$	(11)		3. $\Leftrightarrow \overline{\overline{(a \vee a)} \vee \overline{(b \vee b)}}$	(11)
1. $a \vee b \Leftrightarrow (a \wedge a) \vee (b \wedge b)$	(9‘)	˅ OR	1. $a \vee b \Leftrightarrow (a \vee b) \wedge (a \vee b)$	(9)
2. $\Leftrightarrow \overline{\overline{(a \wedge a) \vee (b \wedge b)}}$	(10)		2. $\Leftrightarrow \overline{\overline{(a \vee b) \wedge (a \vee b)}}$	(10)
3. $\Leftrightarrow \overline{\overline{(a \wedge a)} \wedge \overline{(b \wedge b)}}$	(11)		3. $\Leftrightarrow \overline{\overline{(a \vee b)} \vee \overline{(a \vee b)}}$	(11)

a) Die Äquivalenzumformungen sind hier aus Gründen der Übersichtlichkeit in drei Schritte aufgeteilt, die auch in anderer Reihenfolge auftreten können:

1. Bei der Rückführung auf NAND-Gatter wird die Aussage zunächst als Disjunktion, bei NOR-Gattern zunächst als Konjunktion dargestellt.[3]
   Dazu nutzt man Idempotenzgesetze (9‘) $a \Leftrightarrow a \wedge a$ bzw. (9) $a \Leftrightarrow a \vee a$.
2. Dann folgt der eigentliche Clou, es wird doppelt negiert! Dadurch ändert sich der Wert nicht, da es nach dem Involutionsgesetz (10) eine Äquivalenzumformung ist.
3. Schließlich wird die innere (untere) Negation mit den De Morganschen Regeln "aufgebrochen", damit am Ende nur noch einheitliche Gatter auftreten!

Anmerkung: Die große Bedeutung der De Morganschen Regeln liegt darin, dass Sie die Umwandlung von Konjunktionen in Disjunktionen und umgekehrt ermöglichen.

b) siehe rechte Spalte

3 Hinweis: Bei Verwendung von NAND-Gattern wird die Aussage in die Disjunktive Normal-Form (DNF), bei NOR-Gattern in die KNF=Konjunktive Normal-Form überführt. Aber das soll uns hier nicht weiter beschäftigen, Normalformen werden im Informatikunterricht der Kursstufe behandelt.

3. *In gewohnter Weise*

Hier Aufgabe 2 in der bisher verwendeten Schreibweise der Negation dargestellt. Die doppelte Verneinung ist dann zwar nicht so übersichtlich, dafür ist die Darstellung eventuell vertrauter:

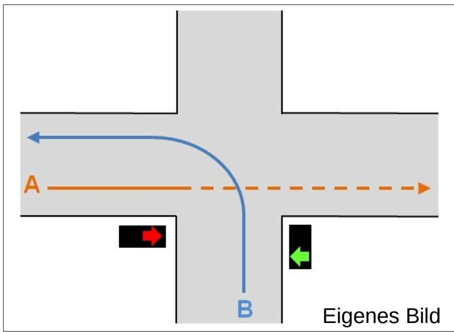
NAND	*Nr.*		NOR	*Nr.*
$\neg a \Leftrightarrow \neg(a \wedge a)$	(9')	¬ *(NOT)*	$\neg a \Leftrightarrow \neg(a \vee a)$	(9)
1. $a \wedge b \Leftrightarrow (a \wedge b) \vee (a \wedge b)$	(9')		1. $a \wedge b \Leftrightarrow (a \vee a) \wedge (b \vee b)$	(9)
2. $\Leftrightarrow \neg\neg[(a \wedge b) \vee (a \wedge b)]$	(10)	∧ *(AND)*	2. $\Leftrightarrow \neg\neg[(a \vee a) \wedge (b \vee b)]$	(10)
3. $\Leftrightarrow \neg[\neg(a \wedge b) \wedge \neg(a \wedge b)]$ [4]	(11)		3. $\Leftrightarrow \neg[\neg(a \vee a) \vee \neg(b \vee b)]$ [5]	(11')
1. $a \vee b \Leftrightarrow (a \wedge a) \vee (b \wedge b)$	(9')		1. $a \vee b \Leftrightarrow (a \vee b) \wedge (a \vee b)$	(9)
2. $\Leftrightarrow \neg\neg[(a \wedge a) \vee (b \wedge b)]$	(10)	∨ *(OR)*	2. $\Leftrightarrow \neg\neg[(a \vee b) \wedge (a \vee b)]$	(10)
3. $\Leftrightarrow \neg[\neg(a \wedge a) \wedge \neg(b \wedge b)]$ [4]	(11)		3. $\Leftrightarrow \neg[\neg(a \vee b) \vee \neg(a \vee b)]$ [5]	(11')

Anmerkung zu den Fußnoten 4 und 5:
Beide Schreibweisen in Aufgabe 2 und 3 haben noch den Nachteil, dass die NAND-Operation (gemäß ihrer Definition) mithilfe der Negation und Konjunktion dargestellt werden muss. In dieser Einheit reicht das für einen ersten Zugang völlig aus, falls man sich aber weitergehend mit der Logikschaltungen beschäftigt, macht es Sinn, ein eigenes NAND-Symbol wie den Sheffer-Pfeil (Pfeil nach oben für NAND) bzw. den Pfeil nach unten (für NOR) zu verwenden. Man erkennt dann sofort, ob in der Verknüpfung nur noch ein Junktor-Typ auftritt und die Schaltung mit nur einem Gatter realisiert werden kann.
Das Dualitätprinzip der Aussagenlogik ist übrigens auch in der Tabelle oben zu erkennen. Durch Vertauschen von ˄ und ◂ in einem der NAND- oder NOR-Gesetze kommt man zum diagonal gegenüberliegenden Gesetz. Noch deutlicher sieht man es in den "dualen" Fußnoten 4 und 5, in denen nur der Pfeil "gedreht" werden muss.

4. **Ampelschaltung**

a) siehe rechts, es ist die Negation der Konjunktion (NAND-Junktor) dargestellt.

b) In der Gleichung $s \Leftrightarrow \neg(a \wedge b)$ können a und b als Eingangswerte und s als Ausgangswert interpretiert werden. Zu allen möglichen Belegungen von a und b ("Vollkonjunktionen") zeigt der Wert von s, ob die Kreuzung sicher ist oder nicht. Unsichere Belegungen werden so durch den Ausgangswert 0 "geblockt".



Eigenes Bild

a	b	*sicher?*
0	0	**1**
0	1	**1**
1	0	**1**
1	1	**0**

Sichere Terme:
$\neg(a \wedge b)$
$\neg a \vee \neg b$
$a \rightarrow \neg b$

c) Es gibt unendlich (!) viele verschiedene sichere Schaltungsgleichungen:

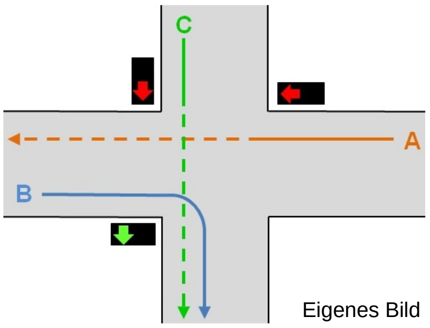
in Normalform, z.B. $s \Leftrightarrow (\neg a \wedge \neg b) \vee (\neg a \wedge b) \vee (a \wedge \neg b)$ oder $s \Leftrightarrow (\neg a \vee \neg b)$
als NAND-Verknüpfung: $s \Leftrightarrow \neg(a \wedge b)$
De Morgansche Regel: $s \Leftrightarrow \neg(a \wedge b) \Leftrightarrow \neg a \vee \neg b$
als Subjunkton geschrieben: $s \Leftrightarrow \neg a \vee \neg b \Leftrightarrow a \rightarrow \neg b \Leftrightarrow b \rightarrow \neg a$
mit Idempotenzen "gedehnt": $s \Leftrightarrow \neg(a \wedge a) \vee \neg(b \wedge b) \Leftrightarrow a \rightarrow \neg b \Leftrightarrow (a \vee a) \rightarrow \neg(b \vee b) \Leftrightarrow \dots$
Doppelte Negationen verwenden: $s \Leftrightarrow \neg\neg(a \rightarrow \neg b) \Leftrightarrow \neg\neg(\neg a \vee \neg b) \Leftrightarrow \dots$

4 Mit dem "Sheffer-Pfeil nach oben" notiert: Konjuktion $a \wedge b=(a\uparrow b)\uparrow(a\uparrow b)$; Disjunktion $a \vee b=(a\uparrow a)\uparrow(b\uparrow b)$.
5 Mit dem "Pfeil nach unten" geschrieben: Konjuktion $a \wedge b=(a\downarrow a)\downarrow(b\downarrow b)$; Disjunktion $a \vee b=(a\downarrow b)\downarrow(a\downarrow b)$.

5. **Kreuzung gesucht**

$s \Leftrightarrow \neg((a \vee b) \wedge c)$

Durch die Negation wird der Fall ausgeschlossen, dass die Fahrspur C gleichzeitig mit A oder B grün hat. Da C weder mit A noch mit B gleichzeitig grün haben darf, ist klar, dass die Fahrspur C die Fahrspuren A und B kreuzt. A und B kreuzen sich dagegen nicht, da keine weitere Bedingung an die Sicherheit gestellt wurde. Eine mögliche Situation sieht man rechts.[6]



Eigenes Bild

a	b	c	s
0	0	0	**1**
0	0	1	**1**
0	1	0	**1**
0	1	1	**0**
1	0	0	**1**
1	0	1	**0**
1	1	0	**1**
1	1	1	**0**

Terme für eine sichere Schaltung:
$\neg(c \wedge (a \vee b))$ oder $\neg c \vee \neg(a \vee b)$ oder $c \rightarrow \neg(a \vee b)$ ….
$(\neg a \wedge \neg b \wedge \neg c) \vee (\neg a \wedge \neg b \wedge c) \vee (\neg a \wedge b \wedge \neg c) \vee (a \wedge \neg b \wedge \neg c) \vee (a \wedge b \wedge \neg c)$

Mögliche Vertiefung: <u>Rechengesetze deuten</u>

Es bietet sich hier auch an, aussagenlogische Terme und Wirkung der Rechengesetze sprachlich anschaulich zu deuten, erste Beispiele hierzu:

De Morgansche Regel (11‘): $s \Leftrightarrow \neg((a \vee b) \wedge c) \Leftrightarrow \neg(a \vee b) \vee \neg c$.
Die Ampeln A oder B dürfen nicht grün zeigen <u>oder</u> Ampel C darf nicht grün zeigen.

Kommutativgesetz (2): $s \Leftrightarrow \neg(a \vee b) \vee \neg c \Leftrightarrow \neg c \vee \neg(a \vee b)$
Die Ampel C darf nicht grün zeigen <u>oder</u> die Ampeln A oder B dürfen nicht grün zeigen.

Distributivgesetz (4‘): $s \Leftrightarrow \neg((a \vee b) \wedge c) \Leftrightarrow \neg((a \wedge c) \vee (b \wedge c))$
Es wird ausgeschlossen, dass A und C gleichzeitig oder B und C gleichzeitig grün zeigen.

Subjunktion: $s \Leftrightarrow \neg(a \vee b) \vee \neg c \Leftrightarrow (a \vee b) \rightarrow \neg c \Leftrightarrow c \rightarrow \neg(a \vee b)$
<u>Wenn</u> die Ampeln A oder B grün zeigen, <u>dann</u> zeigt Ampel C nicht grün.
<u>Wenn</u> die Ampel C grün zeigt, <u>dann</u> zeigen weder A noch B grün.

Ein "sprachlicher" Perspektivwechsel ist möglich, indem man z.B. "nicht grün" durch rot ersetzt:

$s \Leftrightarrow \neg(a \vee b) \vee \neg c$ Ampel A oder B zeigen rot oder Ampel C zeigt rot.
$\Leftrightarrow c \rightarrow \neg a \wedge \neg b$ Wenn Ampel C grün zeigt, dann zeigen A und B rot
...

6 Vgl. Klasse 9, AB "5) Formeln bringen Sicherheit", Aufgabe 2, Datei: M9aug05_logicTraffic_mit4V.odt